



US 20020091670A1

(19) United States

(12) Patent Application Publication

Hitz et al.

(10) Pub. No.: US 2002/0091670 A1

(43) Pub. Date: Jul. 11, 2002

(54) WRITE ANYWHERE FILE-SYSTEM LAYOUT

(76) Inventors: David Hitz, Los Altos, CA (US); Michael Malcolm, Los Altos, CA (US); James Lau, Los Altos Hills, CA (US); Byron Rakitzis, Burlingame, CA (US)

Correspondence Address:

SWERNOFSKY LAW GROUP PC
P.O. BOX 390013
MOUNTAIN VIEW, CA 94039-0013 (US)

(21) Appl. No.: 09/954,522

(22) Filed: Sep. 11, 2001

Related U.S. Application Data

(63) Continuation of application No. 09/153,094, filed on Sep. 14, 1998, now Pat. No. 6,289,356, which is a continuation of application No. 09/108,022, filed on Jun. 30, 1998, now Pat. No. 5,963,962, which is a continuation of application No. 08/454,921, filed on May 31, 1995, now Pat. No. 5,819,292, which is a continuation of application No. 08/071,643, filed on Jun. 3, 1993, now abandoned.

(30) Foreign Application Priority Data

Jun. 2, 1994 (US)..... PCT/US94/06320

Publication Classification

(51) Int. Cl.⁷ G06F 7/00
(52) U.S. Cl. 707/1

(57) ABSTRACT

The present invention provides a method for keeping a file system in a consistent state and for creating read-only copies of a file system. Changes to the file system are tightly controlled. The file system progresses from one self-consistent state to another self-consistent state. The set of self-consistent blocks on disk that is rooted by the root inode is referred to as a consistency point. To implement consistency points, new data is written to unallocated blocks on disk. A new consistency point occurs when the fsinfo block is updated by writing a new root inode for the inode file into it. Thus, as long as the root inode is not updated, the state of the file system represented on disk does not change. The present invention also creates snapshots that are read-only copies of the file system. A snapshot uses no disk space when it is initially created. It is designed so that many different snapshots can be created for the same file system. Unlike prior art file systems that create a clone by duplicating the entire inode file and all of the indirect blocks, the present invention duplicates only the inode that describes the inode file. A multi-bit free-block map file is used to prevent data from being overwritten on disk.

